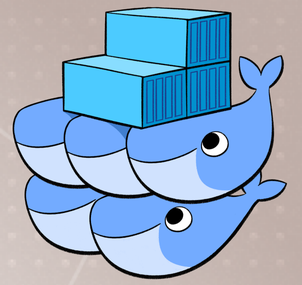
What is Kubernetes?

- Kubernetes = popular container orchestrator
- Container Orchestration = Make many servers act like one
- Released by Google in 2014, maintained by large community
- Runs on top of Docker (usually) as a set of APIs in containers
- Provides API/CLI to manage containers across servers
- Many clouds provide it for you
- Many vendors make a "distribution" of it

Why Kubernetes?

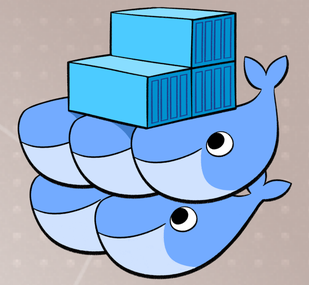
- Review "Swarm Mode: Built-In Orchestration"
- Orchestration: Next logical step in journey to faster DevOps
- First, understand why you *may* need orchestration
- Not every solution needs orchestration
- Servers + Change Rate = Benefit of orchestration
- Then, decide which orchestrator
- If Kubernetes, decide which distribution
  - cloud or self-managed (Docker Enterprise, Rancher, OpenShift, Canonical, VMWare PKS)
  - Don't usually need pure upstream

Kubernetes or Swarm?



- Review "Swarm Mode: Built-In Orchestration"
- Kubernetes and Swarm are both container orchestrators
- Both are solid platforms with vendor backing
- Swarm: Easier to deploy/manage
- Kubernetes: More features and flexibility
- What's right for you? Understand both and know your requirements

Advantages of Swarm



- Comes with Docker, single vendor container platform
- Easiest orchestrator to deploy/manage yourself
- Follows 80/20 rule, 20% of features for 80% of use cases
- Runs anywhere Docker does:
  - local, cloud, datacenter
  - ARM, Windows, 32-bit
- Secure by default
- Easier to troubleshoot

Advantages of Kubernetes

- Clouds will deploy/manage Kubernetes for you
- Infrastructure vendors are making their own distributions
- Widest adoption and community
- Flexible: Covers widest set of use cases
- "Kubernetes first" vendor support
- "No one ever got fired for buying IBM"
  - Picking solutions isn't 100% rational
  - Trendy, will benefit your career
  - CIO/CTO Checkbox